Evaluating Deep Neural Network-based Speaker Verification Systems on Sinhala and Tamil Datasets

S. P. D. Anuraj
Faculty of Engineering
University of Jaffna
Kilinochchi, Sri Lanka
dimuthuanuraj@univ.jfn.ac.lk

S.T. Jarashanth
Faculty of Engineering
University of Jaffna
Kilinochchi, Sri Lanka
jarashanth@eng.jfn.ac.lk

K. Ahilan
Faculty of Engineering
University of Jaffna
Kilinochchi, Sri Lanka
ahilan@eng.jfn.ac.lk

R. Valluvan
Faculty of Engineering
University of Jaffna
Kilinochchi, Sri Lanka
valluvan.r@eng.jfn.ac.lk

T. Thiruvaran
Faculty of Engineering
University of Jaffna
Kilinochchi, Sri Lanka
thiruvaran@eng.jfn.ac.lk

A. Kaneswaran
Faculty of Engineering
University of Jaffna
Kilinochchi, Sri Lanka
kanesh@eng.jfn.ac.lk

***Abstract*— Speaker verification, a biometric identifier, determines whether an input speech belongs to the claimed identity. The existing models for speaker verification have reported performances mainly in English, and no study has experimented with Sinhala and Tamil datasets. This study proposes a semi-automated pipeline to curate datasets for Sinhala and Tamil from videos on YouTube filmed under noisy and unconstrained conditions which represent real-world scenarios. Both Sinhala and Tamil datasets include utterances for 140 persons of interest (POIs) with more than 300 utterances per POI under one or more genres: interviews, speeches, and vlogs. Moreover, this study investigates how domain mismatch affects a speaker verification model trained in English and applied to Sinhala and Tamil. Two deep neural network models trained in English show significant performance drops on Sinhala and Tamil datasets compared to an English dataset as expected due to domain mismatch, however, it is observed that AM-softmax performed better than vanilla softmax. In the future, robust speaker verification models with domain adaptation techniques will be built to improve performance on Sinhala and Tamil datasets.**

Keywords*— *Speaker Verification, Sinhala, Tamil, Dataset, ResNet, Deep neural networks

I. INTRODUCTION

A speaker's voice contains personal traits of the speaker, produced by the unique pronunciation organs and speaking manner of the speakers, such as unique vocal tract shape, larynx size, accent, and rhythm. Therefore, a speaker can be identified automatically via a computer using his/her voice, known as speaker recognition. Major subtasks of speaker recognition include speaker identification and speaker verification. Speaker verification aims at validating whether a speech sample belongs to a claimed identity based on his/her pre-recorded utterances. Moreover, a speaker verification system should be able to cope with intrinsic variations such as accent, dialect, emotion, aging, and speaking manner, and extrinsic variations, such as background noise, music, and reverberation. Commercial applications mainly rely on text-independent speaker verification systems as it is harder to mimic an unknown phrase than a known phrase. Some speaker verification applications include entry control to restricted premises, internet banking, credit card authorization, and access to privileged information.

A large portion of the existing literature on speaker verification systems developed and tested on English datasets, and there is a dearth of studies in other domains. The National Institute of Standards and Technology (NIST) releases English datasets annually to encourage participants to build robust models for speaker verification [1]. However, NIST's datasets are collected manually in controlled conditions. The ICSI [2] and AMI [3] meeting corpora were collected from multi-speaker environments under less controlled conditions. TIMIT dataset performs artificial degradation to mimic real-world noise [4].

SITW [5] is the first dataset that used multimedia data to curate an "in the wild" dataset in unconstrained conditions. Although SITW dataset better represents the real world, it contains utterances for 299 persons of interest (POIs) only due to the difficulties associated with the manual annotation. Nagarani et al. proposed a fully automated pipeline to curate a large dataset under real-world conditions [6]. They collected interview videos of celebrities uploaded to YouTube, shot in challenging multi-speaker acoustic environments, including red carpet, quiet studios, outdoor stadiums, speeches given to large audiences, and others. They released two versions of the dataset: VoxCeleb 1 [6] and VoxCeleb 2 [7]. The VoxCeleb 2 contains utterances for over 6000 POIs, the largest English dataset available to date. Another well-known Chinese dataset called CN-Celeb [8] was released by adapting the pipeline of VoxCeleb with state-of-the-art systems for the sub-components of the pipeline. Unlike VoxConverse, CN-Celeb includes a speech recognition system to double-check whether the output from the active speaker verification is consistent. The final output is manually verified, and the incorrectly marked utterances are discarded. The latest version of CN-Celeb [9] has utterances for 3000 POIs. CN-Celeb also includes 11 variants of genres such as interviews, dramas, advertisements, movies, speeches, vlogs, and others.

No study has experimented with a speaker verification system with neither Sinhala nor Tamil datasets. As a result, we propose a semi-automated pipeline similar to VoxCeleb and CN-Celeb and curate a Sinhala and a Tamil dataset from YouTube videos with background noises, laughter, music, and reverberation. The datasets include 140 POIs for each, where each POI contains one or more of the following genres: interview, speech, and vlog. Here, we name the Sinhala and Tamil datasets as SLCeleb. The subsequent paragraphs

978-1-6654-7607-2/22/$31.00 ©2022 IEEE

discuss several notable works on speaker identification and verification.

Speaker identification is a task similar to speaker verification; however, speaker identification finds the best-matching speaker for an unknown speech from a database of known speakers. Since the systems can be interchangeably adapted to speaker identification and verification, the literature is almost the same.

Pioneering work on speaker identification employed Gaussian mixture models [10]. GMM is a mixture of probability density functions (PDFs) used to model multivariate data. GMM clusters data in an unsupervised way and gives its PDF. GMM-based speaker modeling gives the speaker-specific PDF, in which a probability score is obtained. A decision can be made based on the probability scores of the speaker GMMs. An alternate model was proposed for the speaker verification scheme, called the universal background model (GMM-UBM) [11].

Campbell et al. introduced GMM supervectors addressing the complications in obtaining a fixed number of features from speech samples with variable lengths [12]. This fixed length 'supervector' was then analyzed by machine learning techniques (e.g., support vector machines). Before the deep neural network (DNN) era, factor analysis was applied to compute speaker-dependent and session-dependent GMM supervectors, known as i-vectors [13].

Recently, DNN-based models have revolutionized speaker identification. Unlike conventional methods, DNN models produce highly abstract embedding features from utterances due to powerful feature extraction capabilities. A cutting-edge method for DNN-based speaker identification is x-vectors [14], an embedding generated by training a Time Delay Neural Networks (TDNN). The popular input acoustic features for DNN-based speaker identification include MFCCs and Mel Filter-Banks. Several variants of the x-vectors have been introduced lately: ECAPA-TDNN [15] and Factorized TDNN [16]. As an alternative to x-vectors, several scholars have exploited other DNN architectures, especially VGGNet and ResNet [17].

Chung et al. conducted a phenomenal study on the impact of metric learning-based losses over classification-based losses while training speaker recognition models [17]. They experimented with Softmax, AM-Softmax, and AAM-Softmax for classification losses, whereas triplet loss, prototypical loss, Generalized end-to-end loss, and angular prototypical loss for metric learning objectives.

Several challenges arise in speaker recognition in the wild, such as domain mismatch and noisy problems. Scholars have proposed many domain adaptation and noise reduction methods to overcome these difficulties. DNN-based speaker recognition systems need a large amount of labelled speech data to achieve great success. However, sufficient training data may not always be available for every new application as data annotation is expensive and time-consuming. For example, although large-scale datasets are publicly available for English, there is a scarcity for other languages. Therefore, low-resource speaker recognition should use a large amount of auxiliary data to improve performance. However, this approach faces a domain mismatch between the low-resource data (e.g., Tamil) and auxiliary data (e.g., English). Recently, many DNN-based domain adaptation techniques were proposed to alleviate the mismatch problem: adversarial training-based domain adaptation, reconstruction-based domain adaptation, and discrepancy-based domain adaptation [18].

In addition to the dataset preparation for Sinhala and Tamil, this study examines a speaker verification system trained in English and how well it performs in other domains, such as Sinhala and Tamil. It helps us learn the severity of domain mismatch and develop future solutions to mitigate it. Two models are adopted: speed-optimized residual networks (ResNet) and performance-optimized ResNet. The models were trained on VoxCeleb2 (dev) but not on SLCeleb. Evaluation is done on both VoxCeleb1 (test) and SLCeleb.

The rest of the paper is organized as follows. Section II provides details on how the SLCeleb is curated and the configurations of the speaker verification models. Experiments are given in Section III. Results are discussed in Section IV. Section V concludes the paper with future works.

II. Methodology

This section explains the SLCeleb annotation pipeline and two variants of ResNet-34 [19] architecture for speaker verification, namely, speed-optimized ResNet [20] and performance-optimized ResNet [20]. Our proposed pipeline for speaker verification is shown in Figure 1. An input signal is transformed into Mel Filter-Bank energies and fed into a ResNet model. The ResNet model provides an embedding with a fixed size regardless of the length of the utterance. Initially, the ResNet model is trained for a speaker recognition task. Once the model is trained, the target speakers' embedding is generated and stored in a database. When an unknown person claims an identity, an embedding is generated based on his voice. If the cosine similarity between the embedding and the stored claimed speaker's embedding is above a threshold, it is admitted that the person is the claimed identity.

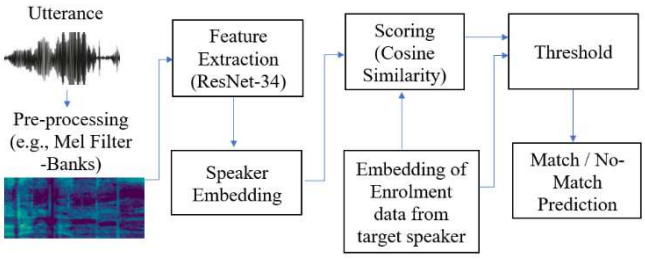

Fig. 1. Our proposed pipeline for Speaker Verification.

A. Speed Optimized ResNet

Initially introduced for image recognition, Residual networks [21] have been adopted for speaker recognition [17]. The speed-optimized ResNet used one-quarter of the channels in each residual block compared to the original ResNet-34 to reduce the computational cost. The number of model parameters in speed-optimized ResNet has been reduced to 1.4 million from 22 million of the original ResNet-34 model parameters. Self-attentive pooling (SAP) [22] aggregates frame-level features into utterance-level representation by giving more weight to the informative frames. This variant is referred to as ResNet-SO. Table I displays the architecture of the ResNet-SO.

TABLE I. ARCHITECTURE FOR THE SPEED OPTIMIZED MODEL. L: LENGTH OF INPUT SEQUENCE, SAP: SELF ATTENTIVE POOLING.

Layer	Kernel Size	Stride	Output Shape
Conv 1	3 × 3 × 16	1 × 1	L × 64 × 16
Res 1	3 × 3 × 16	1 × 1	L × 64 × 16
Res 2	3 × 3 × 32	1 × 1	L/2 × 32 × 32
Res 3	3 × 3 × 64	1 × 1	L/4 × 16 × 64
Res 4	3 × 3 × 128	1 × 1	L/8 × 8 × 128
Flatten	-	-	L/8 × 1024
SAP	-	-	2048
Linear	512	-	512

B. *Performance Optimized ResNet*

The performance-optimized ResNet halves the channels in each residual block on its original ResNet-34, containing 8.0 million parameters. This model is computationally expensive compared to the ResNet-SO as the stride at the first convolutional layer is removed. Channel-wise weighted standard deviation is concatenated with the weighted mean using Attentive Statistics Pooling (ASP) [23] to aggregate temporal features. This variant is referred to as ResNet-PO. Table II displays the architecture of the ResNet-PO.

TABLE II. ARCHITECTURE FOR THE PERFORMANCE OPTIMIZED MODEL. L: LENGTH OF INPUT SEQUENCE, ASP: ATTENTIVE STATISTICS POOLING.

Layer	Kernel Size	Stride	Output Shape
Conv 1	3 × 3 × 32	1 × 1	L × 64 × 32
Res 1	3 × 3 × 32	1 × 1	L × 64 × 32
Res 2	3 × 3 × 64	2 × 2	L/2 × 32 × 64
Res 3	3 × 3 × 128	2 × 2	L/4 × 16 × 128
Res 4	3 × 3 × 256	2 × 2	L/8 × 8 × 256
Flatten	-	-	L/8 × 2048
ASP	-	-	4096
Linear	512	-	512

C. *SLCeleb Dataset Collection Pipeline*

SLCeleb contains over 300 utterances per POI, totaling 140 POIs per Sinhala and Tamil datasets. The proposed pipeline is semi-automated; however, a human check is performed on the output utterances to avoid errors such as an utterance may have multiple speakers or an utterance is incorrectly labeled. There are six stages in the pipeline, and they are discussed below (see Fig. 2).

a) Selecting POIs: We manually selected 140 POIs (per dataset) from Sri Lankan and Indian celebrities in the entertainment, sports, and business sectors or politics or TV personalities. We set a constraint that each POI should have at least an hour of video on YouTube.

b) Download videos and crop portrait images: Videos for each POI are downloaded from YouTube, where each POI may have one or more of the following genres: interview, speech, and vlogs. The videos for each POI are chosen manually, containing intrinsic variations (i.e., factors inherent to the POI – accent, age, emotion, and manner of speaking) and extrinsic variations (e.g., background noise, music, laughter, channel and microphone effects, reverberation) to represent the real-world scenarios. We select YouTube since YouTube is an excellent source for collecting videos with a variety of genres and with the above-said variations and videos are available in abundance.

c) Crop portrait images: The next step is to extract portrait images for each POI. Portrait images are needed to detect and track faces from videos. A portrait image is a matrix that encodes details of a POI's face to identify it on another occasion. It is obtained by clipping the face regions of a POI's various face images and generating a summary image. The Retina Face algorithm [24] is used to obtain the portrait. CNCeleb uses ten pictures for each POI regardless of the videos. Our preliminary experiments revealed that when the pipeline is employed on a video, taking screenshots of the POI from that video at several timestamps increases the precision. The reason is that the face of a POI changes over time, and the POI may wear ornaments, put on makeup, or have different hairstyles, so a global portrait would not work for each video. It is necessary to have local portraits for each video. So, we extract ten face images per POI per video.

d) Face Detection and Tracking: First, detect all faces appearing in each video frame using RetinaFace. Then identify if the target POI appears in each video frame by comparing the faces with the POI portrait. The facial comparison is performed using the ArcFace face recognition system [25]. Later, MOSSE face tracking system [26] is performed to produce face streams.

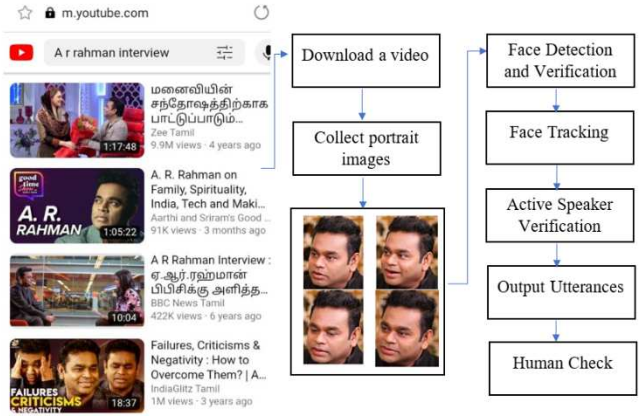

Fig. 2: Data Annotation Pipeline for curating Sinhala and Tamil datasets.

e) Active Speaker Verification: This sub-module is used to verify whether the speech comes from the POI. Sometimes, the POI is on the frame, but the speech comes from another person in the frame or even from someone who is not in the frame. A SyncNet model [27] is applied to check whether the stream of mouth movement and a stream of speech are synchronized. Stream of mouth movement is extracted from the face stream produced by the MOSSE system.

f) Human Check: With the pipeline, our system can crop several utterances of a particular POI for each video. The end of a crop most likely happens when the face of the POI disappears from the frame or the POI stops talking. Once the utterances are collected, the human check begins. The objective of the human check is to resolve discrepancies. This

process is straightforward: select an utterance if it belongs to the target POI only or discard it.

III. Experiments

A. Configurations and Details for the Speaker Verification Model

The audio samples are converted to 64-dimensional log Mel Filter Bank energy features with a window size of 25ms and a shift size of 10ms. The training is done using utterances of at least 3 seconds. During the evaluation, a random sample with the size of 2 seconds from each utterance is extracted.

Evaluations are performed on RestNet-SO and ResNet-PO models with two different loss functions: Softmax and Additive Margin Softmax [17]. We report the results for VoxCeleb1 (test) and SLCeleb.

We adopt a publicly available PyTorch-based Speaker Recognition system [17] and modify it to accommodate speaker verification. Gradients are calculated and updated using categorical cross-entropy loss and Adam optimizer. The initial learning rate of the Adam optimizer is set to 0.001. A scheduler is used to decay the learning rate by 5% after every ten epochs. The models are trained using Nvidia T4 GPU with 16GB memory for 60 epochs with a batch size of 200.

B. Creation of SLCeleb (test) Trial List

Experiments are conducted on Sinhala and Tamil datasets of the SLCeleb, independently. For preparing a trial list for evaluating speaker verification, utterances of 40 POIs per dataset are used. The utterances of the remaining POIs have not been utilized in this paper. One hundred utterances per POI are selected to create the trial list. Moreover, each utterance is duplicated eight times. For example, the trail list for the Sinhala dataset itself contains 32,000 utterances ($40 \times 100 \times 8$). A set of two utterances are selected out of the pool of 32,000 utterances, where duplicate pairs are discarded. Eventually, a set of trail pairs are generated (positive if both utterances belong to the same POI; negative if the utterances are drawn from different POIs). Thus, we created 34,377 positive pairs and 1,511 negative pairs for the SLCeleb-Sinhala test dataset.

C. Evaluation Criteria

The results are reported in two metrics: Equal Error Rate (EER) and minimum Detection Cost Function (minDCF) [1]. Lower values mean better performance for both metrics.

IV. Results and Discussion

Table 3 compares the performances of ResNet-SO and ResNet-PO models on the VoxCeleb1 test dataset. The AM-Softmax achieves a better result than the vanilla softmax loss function. Moreover, ResNet-PO performs better than ResNet-SO.

TABLE III. Performance on VoxCeleb 1 (test) on ResNet-SO and ResNet-PO.

Dataset	Loss	ResNet-SO		ResNet-PO	
		EER	minDCF	EER	minDCF
VoxCeleb1 (test)	Softmax	6.14%	0.4158	5.94%	0.4382
VoxCeleb1 (test)	AM-Softmax	3.52%	0.2243	3.31%	0.3513

TABLE IV. Performance on SLCeleb Tamil (test) on ResNet-SO and ResNet-PO.

Dataset	Loss	ResNet-SO		ResNet-PO	
		EER	minDCF	EER	minDCF
Tamil (test)	Softmax	10.70%	0.5814	10.3%	0.4977
Tamil (test)	AM-Softmax	7.23%	0.3914	6.78%	0.4662

Table 4 and Table 5 compare the performances on SLCeleb (Tamil) and SLCeleb (Sinhala), respectively. The results emphasize the benefits of using AM-Softmax over vanilla Softmax. Vanilla Softmax penalizes classification error, but it cannot inform the model to increase inter-speaker distance (the latent space distance between different speaker embedding) while minimizing intra-speaker distance (the latent space distance between the same speaker embedding). AM-Softmax alleviates this problem due to the addition of cosine margin. Furthermore, the results reveal that ResNet-PO performs better than ResNet-SO. The reason is that the ResNet-SO has significantly lesser parameters than the ResNet-PO and hence loses some fine-grained data.

TABLE V. Performance on SLCeleb Sinhala (test) on ResNet-SO and ResNet-PO.

Dataset	Loss	ResNet-SO		ResNet-PO	
		EER	minDCF	EER	minDCF
Sinhala (test)	Softmax	11.96%	0.6979	10.24%	0.6386
Sinhala (test)	AM-Softmax	7.71%	0.4158	7.29%	0.4929

Since the models were trained solely in English, we observe a substantial performance drop in SLCeleb due to domain mismatch. This problem could be alleviated by training the model on SLCeleb.

V. Conclusions

This paper introduced Sinhala and Tamil datasets to investigate how domain mismatch affects a deep neural network-based speaker verification model trained solely in English. The datasets were curated using unconstrained videos uploaded on YouTube by proposing a semi-automated pipeline. The speaker verification evaluation was conducted on two variants of the ResNet model trained in English (speed-optimized ResNet and performance-optimized ResNet) using English, Sinhala, and Tamil datasets. Both model variants were tested with vanilla Softmax and AM-Softmax loss functions. The following conclusions were drawn: model performs poorly in Sinhala and Tamil than in English, performance-optimized ResNet performs better than speed-optimized ResNet, and AM-Softmax performs better than vanilla Softmax, regardless of the dataset. In the future, we will focus on building a robust speaker verification model with domain adaptation techniques to bridge the performance gap between English, Sinhala, and Tamil datasets.

REFERENCES

[1] S. O. Sadjadi, C. Greenberg, E. Singer, L. Mason, and D. Reynolds, "The 2021 NIST Speaker Recognition Evaluation," in The Speaker and Language Recognition Workshop (Odyssey 2022), Jun. 2022, pp. 322–329. doi: 10.21437/Odyssey.2022-45.

[2] A. Janin et al., "The ICSI Meeting Corpus," in 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)., 2003, vol. 1, no. May, pp. I-364-I–367. doi: 10.1109/ICASSP.2003.1198793.

[3] I. McCowan et al., "The AMI meeting corpus," Proc. 5th Int. Conf. Methods Tech. Behav. Res., vol. 88, no. January, 2005.

[4] John S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren, "Acoustic-Phonetic Continuous Speech Corpus," no. November 1992, 1990.

[5] M. Mclaren, A. Lawson, L. Ferrer, D. Castán, and M. Graciarena, "The Speakers in the Wild Speaker Recognition Challenge Plan," Interspeech 2016 Spec. Sess. San Fr., pp. 818–822, 2016.

[6] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: A Large-Scale Speaker Identification Dataset," in Interspeech 2017, Aug. 2017, vol. 2017-Augus, pp. 2616–2620. doi: 10.21437/Interspeech.2017-950.

[7] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep Speaker Recognition," in Interspeech 2018, Sep. 2018, vol. 2018-Septe, no. i, pp. 1086–1090. doi: 10.21437/Interspeech.2018-1929.

[8] Y. Fan et al., "CN-Celeb: A Challenging Chinese Speaker Recognition Dataset," in ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), May 2020, vol. 2020-May, pp. 7604–7608. doi: 10.1109/ICASSP40776.2020.9054017.

[9] L. Li et al., "CN-Celeb: Multi-genre speaker recognition," Speech Commun., vol. 137, pp. 77–91, Feb. 2022, doi: 10.1016/j.specom.2022.01.002.

[10] D. A. Reynolds and R. C. Rose, "Robust text-independent speaker identification using Gaussian mixture speaker models," IEEE Trans. Speech Audio Process., vol. 3, no. 1, pp. 72–83, 1995, doi: 10.1109/89.365379.

[11] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn, "Speaker Verification Using Adapted Gaussian Mixture Models," Digit. Signal Process., vol. 10, no. 1–3, pp. 19–41, Jan. 2000, doi: 10.1006/dspr.1999.0361.

[12] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," IEEE Signal Process. Lett., vol. 13, no. 5, pp. 308–311, 2006, doi: 10.1109/LSP.2006.870086.

[13] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-End Factor Analysis for Speaker Verification," IEEE Trans. Audio. Speech. Lang. Processing, vol. 19, no. 4, pp. 788–798, May 2011, doi: 10.1109/TASL.2010.2064307.

[14] D. Snyder, "X-Vectors: Robust Neural Embeddings for Speaker Recognition," no. March, 2020.

[15] B. Desplanques, J. Thienpondt, and K. Demuynck, "ECAPA-TDNN: Emphasized Channel Attention, Propagation and Aggregation in TDNN Based Speaker Verification," in Interspeech 2020, Oct. 2020, vol. 2020-Octob, pp. 3830–3834. doi: 10.21437/Interspeech.2020-2650.

[16] D. Povey et al., "Semi-orthogonal low-rank matrix factorization for deep neural networks," Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH, vol. 2018-September, no. 2, pp. 3743–3747, 2018, doi: 10.21437/Interspeech.2018-1417.

[17] J. S. Chung et al., "In defence of metric learning for speaker recognition," Proc. Annu. Conf. Int. Speech Commun. Assoc. INTERSPEECH, vol. 2020-October, pp. 2977–2981, 2020, doi: 10.21437/Interspeech.2020-1064.

[18] Z. Bai and X.-L. Zhang, "Speaker Recognition Based on Deep Learning: An Overview," Neural Networks, vol. 140, pp. 65–99, Dec. 2020, doi: 10.1016/j.neunet.2021.03.004.

[19] H. S. Heo, B.-J. Lee, J. Huh, and J. S. Chung, "Clova Baseline System for the VoxCeleb Speaker Recognition Challenge 2020," pp. 1–3, Sep. 2020, [Online]. Available: http://arxiv.org/abs/2009.14153

[20] Y. Kwon, H.-S. Heo, B.-J. Lee, and J. S. Chung, "The ins and outs of speaker recognition: lessons from VoxSRC 2020," in ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Jun. 2021, vol. 2021-June, no. 2, pp. 5809–5813. doi: 10.1109/ICASSP39728.2021.9413948.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Jun. 2016, vol. 2016-Decem, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[22] W. Cai, J. Chen, and M. Li, "Exploring the Encoding Layer and Loss Function in End-to-End Speaker and Language Recognition System," Speak. Lang. Recognit. Work. ODYSSEY 2018, pp. 74–81, 2018, doi: 10.21437/Odyssey.2018-11.

[23] K. Okabe, T. Koshinaka, and K. Shinoda, "Attentive Statistics Pooling for Deep Speaker Embedding," in Interspeech 2018, Sep. 2018, vol. 2018-Septe, pp. 2252–2256. doi: 10.21437/Interspeech.2018-993.

[24] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S. Zafeiriou, "RetinaFace: Single-stage Dense Face Localisation in the Wild," May 2019, [Online]. Available: http://arxiv.org/abs/1905.00641

[25] J. Deng, J. Guo, J. Yang, N. Xue, I. Cotsia, and S. P. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 2019-June, no. 8, pp. 1–1, Jan. 2021, doi: 10.1109/TPAMI.2021.3087709.

[26] D. Bolme, J. R. Beveridge, B. A. Draper, and Y. M. Lui, "Visual object tracking using adaptive correlation filters," in 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Jun. 2010, no. June, pp. 2544–2550. doi: 10.1109/CVPR.2010.5539960.

[27] J. S. Chung and A. Zisserman, "Out of Time: Automated Lip Sync in the Wild," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 10117 LNCS, no. i, 2017, pp. 251–263. doi: 10.1007/978-3-319-54427-4_